

REMARKS

Applicants and Applicants' Attorney wish to thank the Examiner for the courtesies extended in granting the in-person interview held on September 12, 2008. Applicants submit that the amendments and remarks herein are consistent with and otherwise summarize the substance of the content of that discussion.

The most recent Office Action mailed July 10, 2008 ("*Office Action*") considered claims 1, 4-23, and 26-27. The *Office Action* objected to dependent claim 4 due to an incorrect dependency, and further rejected each of the pending claims under 35 U.S.C. § 102(b) as being anticipated by Jonathan E. Cook, *Supporting Rapid Prototyping Through Frequent and Reliable Deployment of Evolving Components*, 201 IEEE, pp. 194-199 ("*Cook*"). In addition, the *Office Action* rejected claims 18-19, and 23 under 35 U.S.C. § 103(a) under *Cook* in view of the previously discussed *Pratschner* and/or *Eisenbach* references.¹

With this paper, Applicants amend the preambles of independent claims 1, 22, and 26-27, and respectfully traverse the arguments made in the most recent *Office Action*.

As previously discussed in Applicants' prior response, Applicants' invention as generally recited in amended claims 1, 20, 22, 26, and/or 27 relates to a system that automatically and differentially provides target component access to a requesting component, such as based on whether the target components are platform or library components. For example, and as

¹ S. Pratschner, *Simplifying Deployment and Solving DLL Hell with the .NET Framework*, Nov. 2001, Microsoft Corporation, pp. 1-12 ("*Pratschner*"); S. Eisenbach et al., *Managing the Evolution of .NET Programs*, FMOODS 2003, LNCS 2884, pp. 185-198, 2003 ("*Eisenbach*").

Applicants reserve the right to challenge the sufficiency of the *Gunderloy* and *Pratschner* references under 102(b) as there may be some question as to the publication date(s), as these documents appear to be only online publications.

described throughout Applicants’ specification, target components that qualify as “platform components” can be provided to requesting components on a dynamic basis based on whatever happens to be at least the earliest version of the platform component that the requesting component can accept. *E.g.*, ¶ 32 of Applicant’s Application Publication. Platform components can also be overwritten by new “versions” of platform components. *Id.* By contrast, “library components” are those components that are typically or rarely overwritten, and generally maintained side-by-side with other components in the system. *Id.* (Applicants have amended the preamble of the present independent claims primarily to emphasize these points.)

Applicants’ invention thus automatically provides access to various versions of library components when the requesting component asks for a specific version of that target component. *Id.*; *see also* ¶¶ 36, 41-42. This differential, hybrid approach to maintaining and providing target components on the system as taught and claimed by Applicants provides a developer and/or administrator with a much greater deal of flexibility and stability in component-component access requests, without necessarily mandating the all-or-nothing approaches found in the cited art.

By contrast, the *Cook* reference cited in the most recent *Office Action* fails to teach or describe each element recited in Applicants’ claims. For example, the *Office Action* cites the *Introduction* section of *Cook* for the limitations of claim 1. In this section, *Cook* discusses a problem in the art of “knowing what changes were made between successive versions of components, and knowing which versions of which components are compatible.” This passage, however, refers to a generalized statement of a problem in the art that *Cook* is *trying to overcome*, and fails to state how or if those goals are accomplished.

For example, the first limitation of Applicants' claim 1 essentially recites:

- 1) receiving a request from a requesting component for access to a target component;
- 2) wherein the request includes an indication of the lowest possible version of the target component that the requesting component can accept.

By contrast, *Cook* teaches *simultaneous different versions* of the same component, which are used at the same time since each successive component version solves a particular problem evident in a prior version of the component. Thus, if component A had versions 1, 2, and 3, the *Cook* reference teaches that each version effectively forms part of the same component, albeit within a different "domain." See *Cook*, pg. 195 ("Running a specified set of these versions, rather than just the latest one, can increase the reliability of the entire system throughout the evolution process.")

Accordingly, the most that could be said from these and other passages in the *Cook* reference is that the system taught by *Cook* might receive some request for a target component from a requesting component, and that the system uses multiple different versions of the requested target component during execution to satisfy the request. Absent in any of the discussion in *Cook*, however, is the notion that a requesting components specifically indicates "the lowest possible version of the target component that the requesting component can accept." Thus, for at least this reason, *Cook* fails to anticipate each element of limitation 1 in claim 1, and moreover teaches away from this limitation, rendering the rejection under § 102 moot.

In addition to the foregoing, however, the *Cook* reference, whether singly or in combination with any of the references of record, also fails to teach each element of the fourth limitation of claim 1 (similar limitations are also found in claims 20, 22, 26, and 27). For

example, the fourth limitation of claim 1 recites providing different requesting components with different versions of target components based on whether the target component is a "platform component" (*i.e.*, overwritable with another servicing) or a "library component" (*i.e.*, not overwritable, placed beside other versions) so that:

- 1) if the target is a "platform component," only the most recent "servicing" of the component is used; and
- 2) if the target is a "library component," only the specified lowest acceptable version stated by the requesting component is used.

(*See also* claims 20, 22, 26, and/or 27.) For these limitations, the *Office Action* again cites the Introduction of the *Cook* reference for point "1" above, and further cites page 198 point "2" above. Applicants respectfully submit, however, that neither of these passages cited within *Cook*, nor any other passage in the *Cook* reference teaches or suggests each element of this limitation.

At the outset, for example, *Cook* fails to teach or even mention anything approximating a "servicing" value, much less a "servicing" value that is also used in conjunction with a component or library "version." Rather, *Cook* merely teaches how to deal with multiple different versions of a component.

In addition, the *Cook* reference teaches a system in which multiple different versions of a component are left on the system whenever a new one is installed. Specifically, *Cook* teaches that, over time, once newer versions of the component are deemed to reliably satisfy a request during a testing process, the older versions of the component can be removed from the system. *See Cook*, pg. 195, ¶¶ 1-3. Thus, the *Cook* reference fails to teach or suggest any component that

could be considered strictly a "library" component or a "platform" component, such as recited in independent claims 1, 20, 22, 26, and/or 27, which results in a different application of rules depending on the type of component requested. Rather, loosely analogizing *Cook* to Applicants' invention, the components taught by *Cook* might best be construed as being *both* a library and platform component, where the same rules of addition, access, and removal apply to each component in the system.

Accordingly, Applicants respectfully submit that *Cook* fails to teach each limitation – much less each element of each limitation – of independent claims 1, 20, 22, 26, and 27, and, as such that the claims as previously presented, are allowable over the art of record.

In view of the foregoing, Applicants respectfully submit that the other rejections to the claims are now moot and do not, therefore, need to be addressed individually at this time. It will be appreciated, however, that this should not be construed as Applicants acquiescing to any of the purported teachings or assertions made in the last action regarding the cited art or the pending application, including any official notice. Instead, Applicants reserve the right to challenge any of the purported teachings or assertions made in the last action at any appropriate time in the future, should the need arise. Furthermore, to the extent that the Examiner has relied on any Official Notice, explicitly or implicitly, Applicants specifically request that the Examiner provide references supporting the teachings officially noticed, as well as the required motivation or suggestion to combine the relied upon notice with the other art of record.

The Commissioner is hereby authorized to charge payment of any of the following fees that may be applicable to this communication, or credit any overpayment, to Deposit Account No. 23-3178: (1) any filing fees required under 37 CFR § 1.16; (2) any patent application and

reexamination processing fees under 37 CFR § 1.17; and/or (3) any post issuance fees under 37 CFR § 1.20. In addition, if any additional extension of time is required, which has not otherwise been requested, please consider this a petition therefor and charge any additional fees that may be required to Deposit Account No. 23-3178.

In the event that the Examiner finds remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney.

Dated this 6th day of November, 2008.

Respectfully submitted,

/Michael J. Frodsham/

RICK D. NYDEGGER
Registration No. 28,651
MICHAEL J. FRODSHAM
Registration No. 48,699

Attorneys for Applicant
Customer No. 047973